DOCUMENT RESUME

ED 319 384 IR 014 403

AUTHOR	Cheng, Chin-Chuan
TITLE	Microcomputer-Based User Interface for Library Online Catalogue.
INSTITUTION	Illinois Univ., Urbana. Language Learning Lab.
REPORT NO	LLL-T-8-85
PUB DATE	Oct 85
NOTE	15p.
PUB TYPE	Guides - Non-Classroom Use (055) -- Reports - Descriptive (141)
EDRS PRICE	MF01/PC01 Plus Postage.
DESCRIPTORS	College Libraries; *Computer System Design; *Gateway Systems; Higher Education; Microcomputers; *Online Catalogs; *Online Searching; Search Strategies
IDENTIFIERS	Mainframe Computers; Natural Language Processing; University of Illinois; *User Cordial Interface

ABSTRACT

A microcomputer-based user interface was developed and programmed for the library computer systems at the University of Illinois. Designed to provide user-friendly access to the two components of the online catalog on the library's IBM 3081 mainframe computer, the interface program resides on the IBM PC and queries the user in natural written English. System search commands--which are different for the two components--are then formulated internally by the interface, relieving the user of having to learn two distinct command languages. This description of the interface outlines procedures for conducting author and title, title, author, call number, International Standard Book Number, International Standard Serial Number, and subject searches. The implementation and problems of screen layout, natural language queries, subject searches, and computer programming are discussed in detail. (GL)

****
* Reproductions supplied by EDRS are the best that can be made *
* from the original document. *
****

ED319384

Language Learning Laboratory

LLL

U.S. DEPARTMENT OF EDUCATION
Office of Educational Research and Improvement
EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)
This document has been reproduced as received from the person or organization originating it.
Minor changes have been made to improve reproduction quality.
Points of view or opinions stated in this document do not necessarily represent official OERI position or policy.

LLL-T-8-85
OCTOBER 1985

The School of Humanities

University of Illinois
at Urbana-Champaign

MICROCOMPUTER-BASED USER INTERFACE FOR LIBRARY ONLINE CATALOGUE

BY CHIN-CHUAN CHENG

IR014403

BEST COPY AVAILABLE

"PERMISSION TO REPRODUCE THIS MATERIAL HAS BEEN GRANTED BY
C.C. Cheng
TO THE EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)."

LLL-T-8-85
OCTOBER 1985

MICROCOMPUTER-BASED USER INTERFACE FOR LIBRARY ONLINE CATALOGUE

BY CHIN-CHUAN CHENG

LANGUAGE LEARNING LABORATORY
School of Humanities
University of Illinois at Urbana-Champaign

Technical Report No. LLL-T-8-85

MICROCOMPUTER-BASED USER INTERFACE FOR LIBRARY ONLINE CATALOGUE

Chin-Chuan Cheng

Director of Language Learning Laboratory
Professor of Linguistics and of Chinese
University of Illinois, Urbana-Champaign

October 1985

The views and conclusions contained in this document are those of the author and should not be interpreted as necessarily representing the views or policies, expressed or implied, of the Language Learning Laboratory of the University of Illinois.

Available from: Language Learning Laboratory, University of Illinois at Urbana-Champaign, G70 Foreign Languages Building, 707 South Mathews, Urbana, Illinois 61801

ABSTRACT

The author has designed and programmed a user interface for the library computer systems at the University of Illinois. The interface resides on the IBM PCs which are connected with the database mainframe. It aims at capturing the processes of the library patron's search and at providing a graceful interaction between the user and the library mainframe computer. Once the user initiates a search process by pressing a designated key, the interface actively queries the user in natural written English for appropriate information. Search functions include author-title combination, title, author, call number, international standard book number, international standard serial number, shelf position, and subject searches. The implementation and problems of screen layout, natural-language queries, subject searches, and computer programming are discussed in detail.

HUMAN INTERFACE

The Online Catalogue at the University of Illinois at Urbana-Champaign (UIUC) Library has two distinct components: the Library Circulation System (LCS) and the Full Bibliographic Record system (FBR).[1] The LCS component maintains brief titles and authors of library records and gives holding locations and circulation information in detail. It holds all the records for about 7 million titles in more than two dozen libraries in the State of Illinois. The FBR component maintains bibliographic items in full, as in the card catalogue. It contains about 900,000 records.

LCS is a modified version of the Ohio State University's Library Control System. FBR is the bibliographic subsystem of the Western Library Network (WLN). Although access to both systems from the same terminal has been made possible by implementing them on an IBM 3081 mainframe, the systems maintain two distinct command languages. It is difficult for most patrons to master these two sets of commands. I have designed and programmed a user interface aimed at capturing the natural processes of the user search and at providing a graceful interaction between the patron and the computer. The interface program resides on the IBM PC and queries the user in natural written English. Natural language communication does not require additional learning by the user. In this context, the only requirement is the ability to type responses to a system inquiry. System search commands are formulated internally by the interface. Thus any library user can interact with the microcomputer to search for the desired material.

What are the natural processes of library search and how does one design a graceful human-machine interaction? These are, essentially, the questions that I will discuss in this paper.

SCREEN LAYOUT

The interface is microcomputer-based, and its loading of the program does not depend on the database mainframe. Each IBM PC is loaded individually from cassette or disk. At UIUC, the 60 PCs used as library terminals do not have disk drives, and loading is done with a cassette player. It takes about five minutes to load the program this way. Initially, the interface asks the operator to enter the code of the department library where the PC is located and to determine the setting as to whether patrons are allowed to charge departmental holdings from that PC. The time of day is entered at this point. Then the interface clears the screen, displays the time in hours, minutes, and seconds at the upper right corner of the screen. The following ready prompt is displayed on line 1, and the five choices are shown on line 25, bottom line of the screen:

INTERFACE READY (See the bottom line to search; press <?> for help.)

Press 1 for Author-Title 2: Title 3: Author 4: Call No. etc. 5: Subject

---

1. I would like to thank Dorothy Evans, Robert Hart, Stephen Helmreich, and Michael Gorman for their comments on an earlier version of this writing.

Access to a help sequence is gained by pressing the <?> key. There are two screenfuls of explanations. The first one explains the difference between LCS and FBR. The second one shows the user where to locate the <ERASE>, <ENTER>, and <Esc> key on the keyboard. It warns of the difference between the letter <L> and the number <1>. It also mentions that when the user is asked to "type", the <ENTER> key should be pressed to signal the end of input. At the end of this sequence, the user is taken back to the "INTERFACE READY" prompt.

To begin a search, the user presses one of the keys numbered 1 through 5, as indicated by the bottom-line message. As a number key is pressed, the interface erases the ready prompt, writes "Beginning of Search" in its place, and clears the bottom line. As soon as the type of search is identified through the queries, the message "Author-Title Search", "Title Search", "Author Search", "Call Number Search", "ISBN Search", "ISSN Search", "Shelf Position Search", or "Subject Search" is displayed on the bottom line. The search string entered by the user is also displayed on the bottom line. This bottom-line display remains there, with necessary updates as the search progresses, until the end of this round of search. The queries, answers, and mainframe responses continue to fill the screen until line 24; at that point further display causes the screen to scroll up. Thus the user can always review the last portion of the interaction to assist with the next search.

The interface-initiated queries and the strings entered by the user are shown in high intensity and records obtained from the mainframe are displayed in low intensity on a monochrome monitor. With a color monitor, high intensity remains high-intensity white, except that user input is green, and low intensity is shown as yellow lettering against a blue background. The different types of information are thus color-coded to facilitate easy reading.

Queries and answers that are useful for review remain on the screen until they are scrolled off. Lines showing options from which the user can make a selection are erased as soon as a choice is made. The user never sees a left-over question on the screen. Because of the scrolling, a casual on-looker of a search in progress may feel that the screen is cluttered. Indeed, such comments have been made, and a uniform screen layout suggested. However, on the basis of my own observations of users in action, I feel that the screen display of the interface is not at all confusing. If an interface of this sort is engaging, then the user can continue to interact with the computer. To be engaging, the interface has to capture the thought processes of the user. This ability, of course, depends largely on the functions the interface provides.

SEARCH FUNCTIONS

The functions an interface can provide depend to a large extent on the capabilities of the database system on the mainframe. Our interface combines LCS and WLN (FBR) and offers the search capabilities listed below. For each type of search the queries are listed to show the process

of interaction. Some of the queries are disjunctive and are not necessarily presented in the same sequence. Also notice that related queries are presented one at a time so that users can respond.

1. Author and title search

    a. Type the last name of the author and press <ENTER>:

    b. Type the first important word of the title and <ENTER>:

2. Title search

    a. Is this title a periodical? (Press <Y> if "YES", <ENTER> otherwise.):

    b. Type the first important word of the title and press <ENTER>:

    c. Type the second important word of the title and <ENTER>:

    d. What year of this periodical? Press <ENTER> if unknown:

    e. What volume of this periodical? Press <ENTER> if unknown:

3. Author search

    a. Press <C> if the author is a corporate body (association, institution, government, conference, etc.) Press <ENTER> if the author is a person.

    b. Type the name of the corporate body and then press the <ENTER> key:

    c. Type the last name of the author then press <ENTER>:

    d. Type the first name of the author and <ENTER>:

4. Call number, ISBN, or ISSN search

    a. Press one of these keys:
    <C>: Call Number
    <I>: International Standard Book Number (ISBN)
    <S>: International Standard Serial Number (ISSN)

    b. Type the complete call number and then press <ENTER>:

    c. Type the ISBN:

d. Type the ISSN:

5. Subject search

a. Press <ENTER> to search subject or <C> to search call number in subject area:

b. Type the call number (whole or part) and press <ENTER>:

c. To begin, type a term as general as possible to describe the subject:
(You'll be asked to provide more specific information in a moment.)

In the listing above, not all the subject search queries are shown. Shelf position search is listed under subject search, and query b is to prepare the command that displays LCS records according to the order of call numbers. Subject searches are much more complex and will be treated more fully later. Most of the types of the searches belonging to LCS have already been discussed in an earlier paper.[2] Here I shall discuss how the interface combines LCS with the full records in the WLN system.

Initially, the interface issues commands to the mainframe to search LCS for author, title, author-title combination, call number, and shelf position (subject area call number). Apart from call numbers, LCS records are indexed with some combination of two words from the author and/or the title. This is why the interface asks for two words for each of these searches. If LCS finds more than 10 records, the interface then asks the user to provide a third word to match with the records obtained. Searches on LCS may result in more than 40 matches, in which case the interface offers an option to the user of searching full records immediately in the hope that WLN will find more specific records. The user may choose to look at all the retrieved records regardless of the number of items found. When all the retrieved records have been displayed, the interface offers an option to search full records. If no items are found on LCS, the user is also offered an option to search full records before searching the other libraries on LCS.

The LCS search strings often need to be elaborated for full record searches with WLN. If necessary, the interface queries the user for more information. For example, a search of the author-title combination for LCS requires only the last name of the author and the first word of the title. If the search reaches WLN and no matching records are found, the interface presents the user with:

More information is needed to narrow the search.
Type the first name of the author and <ENTER>. Press <ENTER> if none:

Type other words of the title. Press <ENTER> if none:

---

2. See Chin-Chuan Cheng and Kurt R. Murphy, "The IBM PC as a Public Terminal on LCS", *Information Technology and Libraries* 3.1:62-68 (March 1984).

A search that starts out with LCS may return to LCS from WLN so that other libraries can be searched. The user is, of course, shown a message indicating whether the circulation records or full records are being searched.

The subject search is the most important function of WLN. It is discussed below. The interface also provides other functions. It reverts to standard terminal processes if the user does not press one of the designated numerical keys (1 though 5). In this way the user can bypass the interface and issue LCS or WLN commands directly. The interface also allows the library staff to switch to a flashing message to indicate that the mainframe computer is not available. Moreover, when the interface has not been used for four minutes, it clears the screen and displays "Press <B> to Begin" at various locations to save the screen from burn out.

SUBJECT SEARCH

The subject search component is the most complex part of the interface. As patrons may not know the exact wording of the subject in question, some tolerance and intelligence must be built into the program. We approach this problem by trying to help the user sort out the subject by asking for the general area of interest:

> To begin, type a term as general as possible to describe the subject:
> (You'll be asked to provide more specific information in a moment.)

Then we seek to narrow the search with the following sequential queries:

> Type a more specific word or phrase and <ENTER>.
> (Press <ENTER> if you aren't sure of the wording.)

> Press <ENTER> for all or press the corresponding key if the subject is about a topic--<T>, person--<P>, corporate--<C>, or geographical area--<G>.

Once these pieces of information are obtained, the interface displays the following message:

> Searching the Full Bibliographic Records of the holdings acquired since 1975.

The interface determines the type of search command to issue to the mainframe on the basis of the presence or absence of the "specific phrase" asked for in the second query. If that information is absent, a "browse" command is forwarded. If it is present, then a search of the exact subject heading as a string consisting of the "general term" as the heading and the "specific phrase" as the subheading is sent to the mainframe. If an exact heading is not found, the following modifications in the search string are made and searches are attempted one at a time in the order given:

1. The hyphen sign, if present in the user's input, is deleted. This is an idiosyncrasy of the mainframe-terminal communications. For example, "bird-song" in the authority file can only be matched by "birdsong".

2. A truncation sign is added at the end of the search string. This is an attempt to find similar headings.

3. A "browse" command is issued to find alphabetically similar headings. The browse command will eventually find something. However, if the result of this search is not satisfactory to the user, the remaining alternatives are attempted one at a time.

4. The "general term" and the "specific phrase" are strung together as one heading string, and the search is made without the subheading string.

5. Each word in the input string is taken as a heading and the boolean "and" is inserted in between.

6. The search string is taken as a title and a title search is made. This is no longer a subject heading search. However, useful results are often found in this attempt. This type of search makes the interface intelligent to some extent. It will be further discussed later.

During the search of any of these possibilities, if some headings are found, then the user can search the related bibliographic records further. If all the possibilities have been exhausted, the interface indicates this and returns to the "INTERFACE READY" prompt for the user to begin another search.

Of all these possibilities, the boolean search (number 5 given above) is most time-consuming and usually exceeds the limit of the number of intermediate matches imposed by the computer system before a meaningful result is obtained. It is very likely that we will eliminate this type of variant search in the near future. Currently, the interface makes use of the boolean "and" operation for the author/title combination and for alternative 5 of the subject search. It does not use boolean "or". In the past, several attempts were made to incorporate these boolean operations, but the limit of intermediate matches was often reached prematurely. This limitation of the interface is therefore largely a reflection of the characteristics of the WLN programs and database system.

The title search to find relevant headings in alternative 6 is an attempt to implement the thought process and skill of a seasoned library user. If the user is interested in looking at the full record of an item with a title similar to the search string, the full record is searched and the first assigned subject heading appearing there is used as the search string to find the correct relevant headings. This is exactly what a helpful librarian or a skilled catalogue user does. For example, if the user enters "Chinese linguistics" as the subject to be searched, the interface will issue a

"browse" command as outlined in alternative 3 and find "Chinese literature", "Chinese literature--Indexes", etc. as the closest headings. It offers the following options:

> These are the closest subject headings. Do one of the following:
> Type a number and <ENTER> to see the corresponding bibliographic records.
> Press <B> to browse--to see more headings.
> Press <E> to end this search.
> If these headings are not satisfactory, press <ENTER> to try another search.

When the user presses the <ENTER> key alone to indicate interest in trying another subject search, the interface will try alternatives 4 and 5. If no headings are found, then it displays the message "No items were found" and goes on with the title search. At this point the message is shown: "Trying to find CHINESE LINGUISTICS in titles..." In this case, a title "Chinese linguistics: a selected and classified bibliography" is found and the full record displayed. The full record shows "Chinese language--bibliography" as the first assigned subject heading. The user is asked to press <H> to make another attempt to find relevant headings or press <ENTER> to go on. When the <H> key is pressed, the interface extracts the assigned heading from the full record, eliminates the subheading, sends another command to search the subject-heading authority files to the mainframe, and displays "A relevant heading is: Chinese language". The following search results are displayed:

```
+      2. Chinese language
*      1.   --Alphabet
       4.   --Dialects
       5.     --Hainan
       6.     --Hakka
       7.     --Lung-tu
       8.     --China
       9.       --Nan-tung
+     10.   --Etymology
       3.   --To 600
```

The interface examines the headings to see if special symbols are present. In this case, the "+" and "*" signs are attached to some headings. The interface indicates that the user can ask for an interpretation of the symbols. The "+" sign is explained as "There are 'see also' cross references attached to this heading" and "*" as "This heading is invalid. A correct heading is attached." If the user selects a number which has the "+" sign attached, the interface retrieves and displays the "see also" headings. The user is advised to search some of them for relevant items later. The search continues with the selected heading. On the other hand, invalid headings are discarded by the interface and attached correct headings are used to search bibliographic items.

The user may choose one of these headings to view the related bibliographic records. Naturally, the user can also select a heading to start browsing. In this case, the user started out

with a term which is not a heading in the authority file and ends up with correct subject headings. Currently, automated discovery of the correct relevant subject headings takes place only when a full record is retrieved. If a title search retrieves more than one record, only short records are displayed, and if the user does not request to see a full record, there is no way to discover the relevant heading. Furthermore, only the first assigned heading is extracted as the relevant heading to the search in progress. It may turn out that the other assigned headings are more relevant. The user should at least be given a chance to make a decision as to which one is closest to the subject in question. This matter will be considered in future modifications of the programs.

A successful search, no matter whether the starting point is a subject or a title or an author, will result in the display of bibliographic records. If only one item is found, the full record is displayed. If more than one item is found, a maximum of five records in a short form are shown one at a time. For example, if 22 records are found for a subject, the interface shows the first five short records and offers the following options:

> These are short records number 1 - 5. Do one of the following:
> Press <C> for circulation information.
> Type a number and <ENTER> to see the corresponding full record.
> Press <E> to end record display.
> Press <H> to return to heading display.
> Press <ENTER> without a number to see the remaining records.

If <C> is pressed for circulation, the interface asks for a line number to identify the record. Then the circulation (LCS) record is retrieved. The user can charge, renew, or place a save request if the item has already been charged out by someone else. Similarly, if a full record is displayed, the interface asks the user:

> Wish to see circulation information? (Press <Y> if "YES", <ENTER> otherwise.)

At the end of the circulation record display or full record display, the interface reminds the user:

> Earlier we found 22 records.
> Wish to see these short records again? (Press <Y> if "YES", <ENTER> otherwise.)

When the user goes back to the short record display, the other records are displayed. As shown in the options given above, it is also possible to return to subject headings if a subject search is the starting point. At that moment, the user can review retrieved headings and search for other relevant bibliographic items. In summary, the process of search starts with provisional subject terms entered by the user; the user then can: (a) make appropriate selections of the headings; (b) find short or full records; (c) see circulation records for desired action; (d) return to short records display; and (e) return to the subject headings to find other relevant items. The user can terminate the search any time.

PROGRAMMING

In terms of programming, the major part (about 55K bytes) of the interface was written in interpretive BASIC, and a small portion (slightly over 1K bytes) was written in Assembly language. The assembled machine code is loaded beyond the 64K bytes of the BASIC work area. It handles: (a) asynchronous communications using interrupt; (b) clock-display with ticking seconds; and (c) the disabling of the break keys so that the running of the interface cannot be terminated. The rest of the interface functions are performed by the BASIC program.

There were two major considerations in deciding to use BASIC as the programming language. First, the majority of the IBM PCs at the UIUC Library for public use are equipped with a minimum hardware configuration. Currently, there are about 60 public PCs for public use. Each PC has a memory of 128K bytes and an interpretive cassette BASIC in the ROM. No other high-level languages or operating systems are available on these machines. Second, it is much easier to develop and test the interface using an interpretive BASIC with immediate feedback. I have used the IBM PC, XT, and AT at different times to program the interface.

The library PCs are dedicated to library searches only; no other use is allowed. Since the interface is reloaded only when a revision is made or after inadvertent interruptions such as power failures, the use of the slower loading cassette is not a problem. Moreover, the lack of disk drives discourages attempts by students to use these dedicated machines as word processors or calculators.

Is BASIC fast enough to perform all the required functions? Yes. The two speed-critical areas of communications and time display are handled by the assembled machine code. Time display is interrupt-driven. Communications are also interrupt-driven. However, the code is so short that these two functions do not compete seriously and thus cause deterioration of performance. The communications code collects incoming characters and stores them in a buffer for the BASIC routines to receive at their own slower pace. It is thus possible to set the speed of the communications line as high as 9,600 baud without affecting the performance of the interface. For the display and processing of written dialogs, the speed of the interpretive BASIC is more than adequate. The only perceptible delay is that of the mainframe search and not of interface processing.

The version of BASIC we use has a maximum of 64K bytes of program length. In fact, as certain areas of the program segment are used by the system, a 60K-byte program approaches the maximum. The interface almost reaches that upper limit, and future improvements such as those incorporating more intelligence will be somewhat hard to implement due to memory limits. There are, naturally, other BASIC systems that can accommodate longer programs, but the cassette version in the ROM of the IBM PC must be used. We are therefore faced with the difficulty of future expansion. However, this difficulty can be solved by transferring some BASIC routines to the machine code area to reduce program length.

CONCLUDING REMARKS

Library computer systems for patrons must be interactive and user-friendly. To achieve this goal we have created and installed this interface to carry on a human-machine dialog in natural written language. Natural language is an easy form of communication, and, therefore, the user does not need to be specially trained to interact with the computer. Natural language interactions between the mainframe and the user would add a great load to the network communications line and the central processing system. A microcomputer-based interface such as ours avoids such a load and allows the mainframe to perform database functions efficiently.

Moreover, the microcomputer-based interface can be developed and modified independently of mainframe operations. Such independence allows its author to improve and update any part at any time without any logistic constraints. Thus it is possible to exercise a great deal of individual creativity and to incorporate suggestions from users.

Furthermore, as all the commands sent to the mainframe are always formed correctly by the interface, user errors in direct command mode are eliminated, which, again, reduces the load on the mainframe.

Naturally, reduction of the load on the mainframe is only a side effect. The most important function of a microcomputer-based interface is to make the machine more transparent to the user and more intelligent in its performance. We feel, with considerable experiental evidence, that our interface has taken a significant step in this direction.